Cognitive-level Support for Improvisation in Emergency Response

David Mendonça
New Jersey Institute of Technology
mendonca@njit.edu

Yao Hu
New Jersey Institute of Technology
yh44@njit.edu

Qing Gu
New Jersey Institute of Technology
qg3@njit.edu

ABSTRACT

Improvisation—serial and purposeful creativity, exercised under time constraint—is an intensely cognitive endeavor. Accordingly, supporting improvisation requires an understanding of the underlying cognitive processes and an identification of opportunities for support. This paper reports on the development of cognitively-grounded computer-based support for improvisation in a simulated emergency response situation. The application is a computational model which attends to traces of group decision processes, analyzes them, and attempts to achieve fit between its own intentions and those of the group. The current architecture and functioning of the model are discussed, along with an overview of the simulation platform. Current and future workin the areas of model validation and evaluation is described. The results of this work strongly suggest that model-based support for improvisation is possible, but that for the time being will be restricted to synthetic situations, of the kind often used in training exercises.

Keywords

Cognitive modeling, decision support, improvisation.

INTRODUCTION

Extreme events such as natural or technological disasters challenge society's capabilities both for planning and response. Information technologies and advanced modeling techniques continue to expand how society can limit and manage extreme events. Yet, as is evident from the long history of extreme events, a key both to successful planning and effective response is an ability to improvise. This paper reports on the design and development of a computer-based system for supporting improvisation. Improvisation is first discussed as a combined cognitive and behavioral activity within the domain of emergency response, resulting in a set of requirements for systems to support this activity. The design and development of a system to provide such support is then reviewed, along with illustrations of system functionality and anticipated impact on human cognition in a simulated emergency environment. Finally, future work in system refinement, validation and testing is briefly described, along with other opportunities for future research.

BACKGROUND

Supporting improvisation in response to extreme events can mean providing education and training before the event's occurrence or decision support during the response to it. The former approach is akin to planning for improvisation (Kreps, 1991, Weick, 1993, Weick, 1998, Mirvis, 1998). Because emergencies are infrequent, emergency response personnel must learn how to process and reason about new, possibly never-before-seen, information quickly and accurately. Regular training exercises which induce situations requiring creative thinking under time constraint may support personnel in succeeding in such situations—not by training in how to recognize the need for and execute standard operating procedures, but by training in how to recognize and respond to unplanned-for contingencies. Prototype technologies (Mendonça et al., 2003, Beroggi et al., 2001) may be used in tandem with training to provide support for learning.

While research on how to provide cognitive-level technological support for decision making is needed (Zmud, 1986), it is rare (Chen and Lee, 2003). One barrier is that, in order for such support to succeed, its design should follow from an understanding of the cognitive processes that underlie decision making (Lerch and Harter, 2001)—something that is often difficult to accomplish in practice. Group decision making processes have often been characterized by adapting stage models of individual decision making. For example, Simon proposed a four-stage model of decision making, consisting of intelligence, design, choice and revise (Simon, 1977). Many other models may be seen as elaborations or condensations of this model. For example, the decision making process has also been considered as involving two successive stages: consideration set formation and final choice selection (Levin, Huneke & Jasper, 2000). Another stage model of group decision making consists of three phases (Blumberg, 1994): collecting relevant information, generating and evaluating multiple alternatives, and selecting a course of action. In these models and others (e.g., (Levin, Huneke & Jasper, 2000)), activities of information seeking and handling (i.e., foraging) take place mainly in the early stages, and information seeking happens more often in the early stages than information handling. How information is distributed, sought and handled in pre-choice stages constrains choice, since decision makers can only choose from alternatives they know about.

Improvisation in emergency response can be regarded as a two-stage process (Mendonça, forthcoming) In the first stage, the responding organization recognizes either that no planned-for procedure applies to the current situation or that an appropriate planned-for procedure cannot be executed. In the second stage—given that the need to depart from planned-for procedures has been recognized—the organization must assemble a new solution. Due to time constraint, opportunities for revising or revisiting candidate solutions are likely to be limited.

The development of decision support for improvisation benefits from a number of prior advances in the decision sciences. These range from algorithms for logistics to support for visualization and rich communications. A challenge for those developing technology-based tools, however, is that a situation requiring an improvised approach has—almost by definition—not been planned-for. As a result, the set of paths between the current and target state of affairs (i.e., between the current problem state and the goal state) must be generated by the decision maker, perhaps with assistance from a computer-based system. Accordingly, the intelligence embedded within the system must be capable of identifying possible new problem states and evaluating the likelihood that any path between a set of states will lead to a solution (Mendonça and Wallace, forthcoming).

Researchers have long noted the difficulties in evaluating the efficacy of support for solving ill-structured problems. The perspective taken here is that technologies for supporting improvisation are best tested first under conditions which enable collection of a variety of data and comparison of multiple experiment results. These conditions may be created within training situations (Mendonça et al., 2006). For example, operational exercises typically depend on information technology when there is an expected role for the technology in executing the skill in practice. Because operational training is usually conducted in real-time, evaluation and logging tools will play a major role in post-exercise analysis. Logging and virtual reality technologies can be used for the different operational training platforms, but the complexity of the systems—and therefore the development costs—increases with exercise scale.

Various techniques for training for improvisation have been identified, along with a consideration of how information technology might be best employed to support the delivery of these techniques (Mendonça and Fiedrich, 2006). Here, we consider cognitive shadowing, a technique which requires the trainee to put him/herself within the head of a person participating in the exercise. Trainees may then compare their decision making processes with those of another exercise participant. Cognitive shadowing may therefore help develop the capability of response personnel to identify when and how to depart from planned for procedures. Seen from another perspective, the ability to model the cognitive shadowing process ought to lead to a better understanding of how to provide effective training for improvisation. The following sections discuss the design of a computational model that attempts to shadow one or more human decision makers in a simulated emergency response environment.

Decision support for improvisation in emergency response

As suggested by the previous discussion, one approach to supporting improvisation via cognitive shadowing is the integration of computational models of cognition into a decision technologies (Lerch and Harter, 2001, Mendonça and Wallace, 2002, Krishnan et al., 2001). A computational model of cognition (often called simply a cognitive model) may be described as a theory of human cognition that is executable on a computer (Cohen, 1989, Polk and Seifert, 2002). Because computational models attempt to model human cognitive processes, they have the potential to provide powerful means of simulating (Lin and Carley, 2001, Lin, 2000) or supporting human cognitive processes

(Gonzalez et al., 2003, Lerch and Harter, 2001, Jones and Jacobs, 2000, Jones and Mitchell, 1995), particularly when the task is characterized by uncertainty (Montazemi and Gupta, 1997). Once such a model has been expressed in computer-executable code, it may be validated by (i) comparing the reasoning processes of the model to those of human subjects (Tambe, 1996, Newell et al., 1962) or, in a setting where human and computer cooperate, (ii) evaluating whether the reasoning processes of the model and the decision makers are mutually and correctly understood (Jones and Mitchell, 1995).

Model-based support for improvisation should address two fundamental questions. At a cognitive level, the question of *when* to improvise may be conceptualized as a categorization problem that may be influenced by a number of factors. Time pressure (Marsden et al., 2002, Moorman and Miner, 1998) and risk, for example, may influence how the choice is made (Smart and Vertinsky, 1977), in part by reducing the inclination to improvise given that the need exists to do so (Weick, 1993). Prior training may lead decision makers to enact strategies based on recognizing characteristics of past problems in the current one (Klein, 1993). Decision support for recognizing when to depart from planned-for procedures must therefore support—and perhaps even encourage—the comparison of the current decision situation with past ones (Mendonça, forthcoming). A key issue is that event severity and time constraint may discourage decision makers from considering that the situation might not have been planned-for. Decision support should therefore be capable of providing alternative views of event-related data, as informed by analysis of observed patterns of group information foraging behavior.

Once the need to improvise has been recognized, the second question is how to accomplish real-time development and deployment of new procedures. The improvised action may range from substitution (e.g., using a school bus to transport injured persons) to the construction of new procedures (e.g., using fire trucks to provide mobile showers following a chemical exposure). At a cognitive level, then, the question of *how* to improvise may be conceptualized as a search and assembly problem (Newell et al., 1962), which may be influenced by factors such as time available for development and deployment of new procedures, risk in the environment and the results of prior decisions.

Mendonça identified a set of requirements for DSS to support extreme event decision making, which may be grouped within high-level descriptions of various cognitive activities (Mendonça, forthcoming), as shown in Table 1. For example, distinguishing routine from non-routine contingencies requires categorization-type activities, as has been discussed by Klein (1993). Based on the previous discussion, for a system to provide support for improvisation, it may be advantageous to observe certain behavioral and thinking processes of human participants. Given the rarity of extreme events—and the difficulties involved in collecting such data from actual responses to them—a simulated environment is useful in initial tests of the efficacy of this support, as well as the validity of the model embedded within the system. The system described in the remainder of this paper takes a model-based approach to providing support for the activities of search, assembly, constraint satisfaction and communication given in Table 1.

Table 1. Requirements for Extreme Event Decision Support Systems

Cognitive Activity	Requirements
Categorization	Recognize the occurrence of unplanned-for contingencies
Search	Retrieve or infer a referent that is appropriate for the situation
Assembly	Generate one or more new procedures that are derived from this referent
Constraint satisfaction	Ensure that new procedures can be executed in a timely fashion
Communication	Communicate and collaborate with human decision makers
Inference	Reason about interdependent physical systems and the models that represent them

SYSTEM DESIGN AND IMPLEMENTATION

The decision support system functions within a simulated emergency environment, implemented as a computer-based system. Following a brief description of this environment, details of the system itself are presented.

Simulation Environment

The simulation environment consists of human decision makers, a simulation engine that tracks and processes group information seeking and decision processes, and a computer interface between decision makers and the simulation (see Figure 1). The simulation draws upon data from an actual case of emergency response concerning a cargo ship fire with an oil spill (Harrald, Marcus and Wallace, 1990). Each group has five roles: one group coordinator (CO) acts as a facilitator and principal communicator with the simulated system; the other four act as the representatives of four particular emergency services: Police Department (PD), Fire Department (FD), Medical Officer (MO), and Chemical Advisor (CA). The task of the group is to allocate resources from various sites to the incident location in order to meet (given) response goals (e.g., control access to the incident location). Group members' courses of action and corresponding goals are the inputs to the simulator; outputs are current degree of goal attainment, status of previous actions, and current event severity. Interaction between the group and the simulator is via a human-computer interface.

The simulation engine is written in Visual C++ 6.0, and converted to Xtras using Macromedia® Director® Xtra Development Kit as a "plug-in" that extends the functionality of system. Two Xtras are developed to calculate the goal attainment and level of severity respectively. The interfaces for the participants are developed in Macromedia® Director® MX 2004. The system is compiled for use over the world-wide web. The system is embedded within a virtual workspace which is implemented in phpBB (an open source software). The workspace links to a mySQL database management system (DBMS), enabling all system logs, group communications, and individual click streams to be captured and stored in a database maintained by the DBMS.

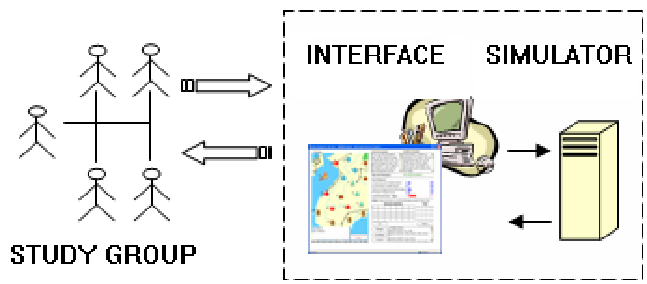

Figure 1: Research Diagram

Three environmental variables may be manipulated within the system, all of which are hypothesized to be relevant to the ability of decision makers to make correct choices about when and how to improvise. The first factor is distribution of information. Common information is information that is accessible to all group members, while shared information is only accessible to some group members. Messages and information about resources may be controlled in order to create conditions where the varying degrees of information are shared. The second environmental variable is time constraint, which is manipulated by changing the speed of the simulation clock. When under low time constraint, the simulation clock is set to go at a normal pace. When under high time constraint, the simulation clock is set to go faster so that groups have less time to complete their tasks. The third variable is level of severity, which is manipulated by changing the initial level of severity and the incremental contribution of various resources (e.g., ladder trucks) to various goals (e.g., control of fire). The coefficient of goal attainment ranges from 0 to 1.

Support Mechanisms

In the simplest case, the need to depart from planned-for procedures arises when the decision specified in the plan is not feasible. More complex situations arise when the planned-for procedure is feasible but is not "best." At present, decisions within the simulation about when to depart from planned-for procedures are made by a human decision maker. In an earlier version of the model presented here (Uschold and Grüninger, 1996) , declarative and procedural knowledge were used in inferring the intention of a responding organization when it is faced with contingency that blocks execution of a planned procedure. The model then produces an improvisation that is consistent with the inferred intention but that is feasible. Declarative knowledge is represented in the form of an ontology and procedural knowledge in the form of a decision logic. The current implementation (written in ACT-R (Anderson et al., 2004)) is being used to develop a more psychologically plausible model, since it is embedded within a cognitive architecture that specifies constraints on human information processing.

The model takes an initial course of action (CA_1) as input, which is assumed to be provided by an emergency response plan or recommended by other group members. CA_1 specifies a sequence of activities to be performed and the goal(s) that the course of action is intended to achieve. The output of the model is an alternative course of actions (CA_2) with reference to the original CA_1, which is also intended to achieve the same goal(s). The scope of this model is therefore to provide an alternative CA based on the initial CA. It does not examine the validity of the initial CA.

The main function of the model is *improv*. It puts the parameters into the slots of the goal chunk before running the model and reads the responses from the slots of the goal chunk upon experiment completion. This function takes four parameters. For the current version of this model, cognition is being modeled within the scenario discussed previously. For example, consider an emergency located at site Z and a goal goal2 related to some CA. Therefore, the destination "Z" and the goal "2" are encoded in the chunk. The decision maker inputs the sequence of actions of the CA. The **mod-chunk-fct** function modifies a chunk. It takes two parameters, one is the name of the chunk to be modified and the other is a list of slot and value pairs. The **chunk-slot-value-fct** function returns the value in a particular slot of a chunk. The parameters it takes are the name of the chunk and the name of the slot. The **buffer-read** function returns the name of the chunk currently in the buffer, the name of which is specified in the parameter.

Each resource has a unique identifier which associates it with a physical object about which decisions can be made. An example of an object is "Fire Truck". A hierarchical ontology to represent the objects. All the objects are under one of the four Object Groups, which are vehicle, boat, equipment and personnel. In the improvisation process, the model only search for alternative resources within same object group. Each object has seven properties. The first property is *role*. It is the role of the emergency response expert, besides the Coordinator, who has access to this type of physical object. The second property is *site*. It is the site where the specific object locates. When the model searches for the alternative resource, it tries to find the alternative resource within the same object group which has the most similarity with the original resource based on object type, role and site. The remaining properties are illustrated in Figure 2, which depicts the ontology of the resources encoded in the model. For example, the object Aerial Ladder Truck is within the object group of Vehicle, and has a role of FD. Other resource specified properties of an object are amount/capacity, driver, and state. Amount/capacity is the amount of certain physical objects in one resource or the capacity of the object given this object is within vehicle or boat object groups. Driver is the availability of a driver for vehicle and boat type objects or the availability of means of transportation for equipment and personnel type objects. State is the current state of the object. It can be either busy or free.

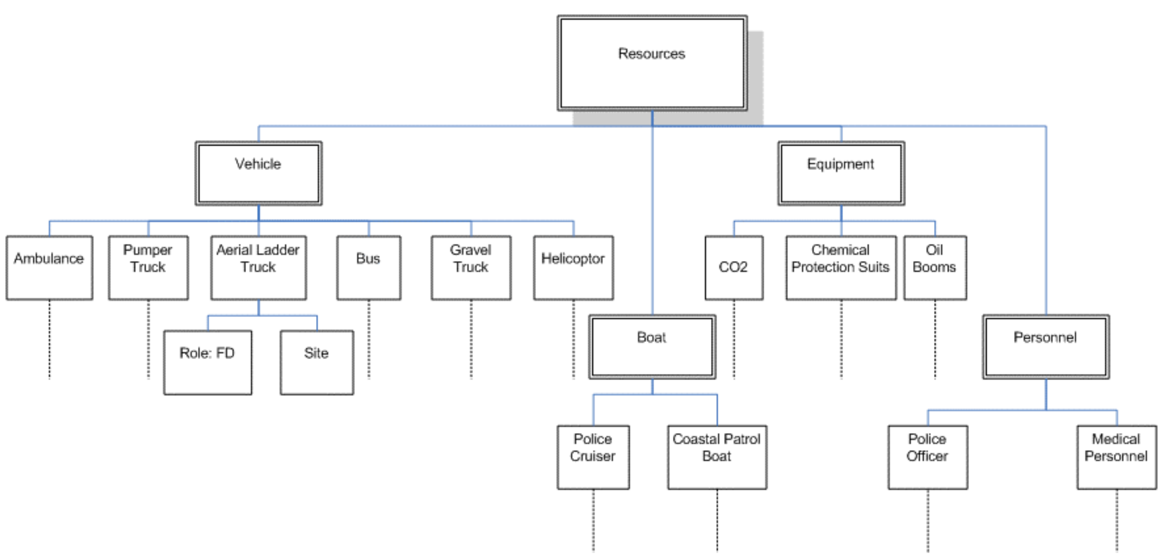

Figure 2: Ontology of the Model

The constraint chunk-type defines constrain and requirement for the objects. In this model, the *slot* slot is set to driver, and the *value* slot is set to 1. This means in order to send the resource to achieve the goals, this resource need to have either a driver (for the vehicle and boat type objects) or means of transportation (for the equipment and personnel type objects). The requirement slot is used to set the object group of the pre-requisite resource when the current retrieved resource does not meet the pre-defined constraint.

There are eight pre-defined modules in the current ACT-R framework, which are declarative memory, procedural, goal, imaginal, audio, vision, speech and motor. Among them, imaginal, audio and speech modules are newly introduced in the current version of ACT-R. The model uses only the cognitive system in ACT-R, and does not rely on any of the perceptual or motor modules. This model uses the imaginal buffer to hold the searching strategy and temporary improvisation states. The imaginal buffer functions as a scratch paper which the model uses to write down the steps and thoughts to support the decision making process.

Illustrative Example

The following example illustrates the functioning of the model. An example input to the model is (improv “ba” “ga” nil nil), which is a symbolic representation of a course of action to "send resource Ba from location B to location G to pick up resource Ga, then send all to destination Z to achieve Goal 2.” In this scenario, the chunk representing resource “Ba” is as follows: (r3 isa resource id "ba" object pumper-truck parent vehicle role fd amount 1 site b driver 1 goal1 n goal2 y goal3 y goal4 n state busy). This means resource “Ba” is a pumper truck (i.e., a vehicle-type resource), and it belongs to the role of fire department. The resource is located at site B. The amount or quantity of this resource is one, and it has driver available. This resource can fulfill goals 2 and 3, which are "Control of Fire at Incident Location" and "Removal of trapped Persons from Danger" respectively. The state of the resource is set to busy, either because the resource is already being used or because of the distance of the site to the destination makes the resource infeasible at the present time.

Since the state of the resource in the original CA is *busy*, the model starts the improving process. In order to find the alternative resource, the model searches through the available resources of the same type with the original resources. It first looks for the resources which have either a driver (i.e., either vehicle- or boat-type resources) or transportation means (for personnel- or equipment-type resources) available. If there are several matches, the model will choose a best one based on the similarity between the original resource and the matched resources. The technique used here is called spreading activation. In this model, the factors taken into consideration are “object”, “parent (type)”, “role”, “site” and “goal.” Each factor has the same weight. The alternative resource which has the most similarity with the original resource in these factors will have the highest activation, and thus will be selected.

In this example, there are five matches after the initial search, which are r2, r7, r8, r9 and r23:

- (r2 isa resource id "ab" object ambulance parent vehicle role mo amount 1 site a driver 1 goal1 n goal2 n goal3 n goal4 y state free)
- (r7 isa resource id "da" object ambulance parent vehicle role mo amount 1 site d driver 1 goal1 n goal2 n goal3 n goal4 y state free)
- (r8 isa resource id "db" object ambulance parent vehicle role mo amount 1 site d driver 1 goal1 n goal2 n goal3 n goal4 y state free)
- (r9 isa resource id "ea" object pumper-truck parent vehicle role fd amount 1 site e driver 1 goal1 n goal2 y goal3 y goal4 n state free)
- (r23 isa resource id "qa" object helicopter parent vehicle role ar amount 1 site q driver 1 goal1 n goal2 y goal3 y goal4 n state free)

Among these matched resources, r9 has the highest activation of 0.18045533. Therefore, resource “Ea” is selected to replace the original resource “Ba” in the new course of action.If there are no matches for the resources with a driver or transportation available, the model will search for the resources without a driver or transportation means. After select the best match among the matched resources, the model will dispatch a driver or vehicle for this alternative resource.

The recommendation of the model is then made available to the human decision makers. Perhaps more importantly, the model provides a detailed trace of its reasoning processes, enabling a comparison between its reasoning processes and those of decision makers who also considered alternatives to the original (but infeasible) course of action. There is considerable effort underway in the ACT-R community to develop automated means for translating these reasoning processes into common written English in order to employ the model as a tool to support the learning process.

DISCUSSION AND CONCLUSIONS

Improvisation in emergency response has been presented as a two-stage process, in which decision makers determine when and how to depart from planned-for procedures. A number of factors may influence how this process is executed. In the emergency response case, three particularly salient factors are distribution of information, time constraint and level of severity. A computer-based platform for investigating how decision making is shaped by variation in these factors has been presented. Support for decision making is provided via a cognitive model, implemented in ACT-R, which takes infeasible courses of action as input and provides feasible (improvised) courses of action as output. Decision makers may then evaluate the proposed courses of action for possible execution. An added benefit of the approach to model development taken here is that the model is meant to explain human cognitive processes during improvisation.

Current work in this program focuses on validation of the model and evaluation of its efficacy in providing decision support. Continued work on model development will enable search and assembly procedures to be influenced by the model observations of decision making (as at present) and information seeking behavior. For example, it is expected that, as time available for implementation decreases, information seeking will become more focused on unique information, even when private information is needed to solve the problem. The main evaluation criterion is the degree of accuracy of the model's understanding of decision makers' reasoning processes. Regarding the efficacy of the system for decision support, one capability of the system is to support reasoning about alternative uses of resources. Use of the model for decision support is therefore expected to increase the amount of effort spent by the group in considering unique information.

ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. National Science Foundation under Grant No. 0449582.

REFERENCES

1. Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C. and Qin, Y. (2004) *Psychological Review,* **111,** 1036-1060.
2. Beroggi, G. E. G., van Gent, D., Mendonça, D. and Wallace, W. A. (2001) In *The International Emergency Management Society Conference 2001: Proceedings*(Ed, Teig, M. B. a. E.) Olso, Norway.
3. Blumberg, H. H. (1994). "Group decision making and choice shift." In Small Group Research: A Handbook, edited by A. P. Hare, H. H. Blumberg, M. R. Davies, and M. V. Kent, 141-154, NJ: Ablex Publishing Company.
4. Chen, J. Q. and Lee, S. M. (2003) *Decision Support Systems,* **36,** 147-160.
5. Cohen, P. (1989) In *The Handbook of Artificial Intelligence*, Vol. 3 (Eds, Cohen, P. R. and Feigenbaum, E. A.) Addison-Wesley, Boston, pp. 1-74.
6. Gonzalez, C., Lerch, F. J. and Lebiere, C. (2003) *Cognitive Science,* **27,** 561-635.
7. Harrald, J. R., Marcus, H. S. and Wallace, W. A. (1990). "The EXXON Valdez: an assessment of crisis prevention and management systems." Interfaces, 20(5), 14-30.
8. Jones, P. M. and Jacobs, J. L. (2000) *IEEE Transaction on Systems, Man, and Cybernetics,* **30,** 397-407.
9. Jones, P. M. and Mitchell, C. M. (1995) *IEEE Transactions on Systems, Man and Cybernetics,* **25,** 1039-1053.
10. Klein, G. A. (1993) In *Decision Making in Action: Models and Methods*(Eds, Klein, G. A., Orasanu, J., Calderwood, R. and Zsambok, C. E.) Ablex Publishing Corp., Norwood, NJ, pp. 138-147.

11. Kreps, G. A. (1991) In *Emergency Management: Principles and Practice for Local Governments*(Eds, Drabek, T. E. and Hoetmer, G. J.) International City Management Association, Washington, D.C., pp. 30-54.
12. Krishnan, R., Li, X., Steier, D. and Zhao, L. (2001) *Information Systems Research,* **12,** 286-301.
13. Levin, I. P., Huneke, M. E. and Jasper, J. D. (2000). "Information processing at successive stages of decision making: need for cognition and inclusion-exclusion effects." Organizational Behavior and Human Decision Processes, 82(2), 171-193.
14. Lerch, F. J. and Harter, D. E. (2001) *Information Systems Research,* **12,** 63-82.
15. Lin, Z. (2000) *Computational and Mathematical Organization Theory,* **6,** 277-310.
16. Lin, Z. and Carley, K. (2001) In *2001 Academy of Management Best Papers Proceedings*Washington, D.C., USA, pp. B1-B7.
17. Marsden, J. R., Pakath, R. and Wibowo, K. (2002) *Decision Support Systems,* **34,** 75– 97.
18. Mendonça, D. (forthcoming) *Decision Support Systems*.
19. Mendonça, D., Beroggi, G. E. G., van Gent, D. and Wallace, W. A. (2006) *Safety Science,* **44,** 523-535.
20. Mendonça, D., Beroggi, G. E. G. and Wallace, W. A. (2003) In *Hawaii International Conference on System Sciences (HICSS-36)*Big Island, HI, pp. 229-237.
21. Mendonça, D. and Fiedrich, F. (2006) *International Journal of Emergency Management,* **3,** 348-363.
22. Mendonça, D. and Wallace, W. A. (2002) In *Hawaii International Conference on System Sciences (HICSS-35)*Big Island, HI, pp. 2882-2888.
23. Mendonça, D. and Wallace, W. A. (forthcoming) *IEEE Transactions on Systems, Man, and Cybernetics: Part A*.
24. Mirvis, P. H. (1998) *Organization Science,* **9,** 586-592.
25. Montazemi, A. R. and Gupta, K. M. (1997) *OMEGA, International Journal of Management Science,* **25,** 643-658.
26. Moorman, C. and Miner, A. S. (1998) *Journal of Marketing,* **62,** 1-20.
27. Newell, A., Shaw, J. C. and Simon, H. A. (1962) In *Contemporary Approaches to Creative Thinking*(Eds, Gruber, H. E., Terrel, G. and Wertheimer, M.) Atherton Press, New York, pp. 63-119.
28. Polk, T. A. and Seifert, C. M. (Eds.) (2002) *Cognitive Modeling,* The MIT Press, Cambridge, MA.
29. Simon, H. A. (1977). "The New Science of Management Decision." Prentice Hall PTR, Upper Saddle River, NJ.
30. Smart, C. and Vertinsky, I. (1977) *Administrative Science Quarterly,* **22,** 640-657.
31. Tambe, M. (1996) In *International Conference on Multi-agent Systems*.
32. Uschold, M. and Grüninger, M. (1996) *Knowledge Engineering Review,* **11,** 93-136.
33. Weick, K. E. (1993) *Administrative Science Quarterly***,** 628-652.
34. Weick, K. E. (1998) *Organization Science,* **9,** 543-555.
35. Zmud, R. W. (1986) In *Decision Support Systems: A Decade in Perspective*(Eds, McLean, E. R. and Sol, H. G.) Elsevier Science, North-Holland, Amsterdam.